

# 特許協力条約



発信人 日本国特許庁（受理官庁）

出願人代理人

特許業務法人 アイ・ピー・エス

あて名

〒 221 - 0052

日本国神奈川県横浜市神奈川区栄町 5 番地 1 横浜クリエーションスクエア 1 2 階

殿

## 手続補正命令書

（法第 6 条、法施 30 条）

〔PCT 3 条（4）（i）、14 条（1）、規則 26〕

発送日（日．月．年）

20.04.2004

出願人又は代理人の書類記号

200310952

応答期間

発送日から 1 月以内

国際出願番号

PCT / JP2004 / 000521

国際出願日（日．月．年）

22.01.2004

出願人（氏名又は名称）

ヒューレット・パッカード デベロップメント カンパニー エル．ピー．

出願人は、上記の期間内に手続の補正をしなければならない。補正すべき事項は次の附属書に示されている。

☐

附属書 A

☐

附属書 B

☒

附属書 C

（注意）

### 補正の方法

手続補正書に補正事項を補正した差替用紙を添付することにより行う。また、手続補正書の「補正の内容」の欄に差替えられる用紙と差替後の用紙との相違について記載する。なお、補正によって書き換えられる用紙の明瞭さ及び複製の可能性に悪影響を及ぼすことなく、手続補正書の「補正の内容」の欄から記録原本に書き換えられることができる性質のものである場合には差替用紙を省略することができる。

（PCT 規則 26.4（a）、法施行規則様式第 15 備考 4 参照）

### 注意

手続の補正がされないときは、国際出願は取り下げられたものとみなす旨の決定がされる。

（法第 7 条第 1 項、PCT 規則 26.5 参照）

この補正命令書及び附属書の写しは、国際事務局

☐

及び国際調査機関に送付した。

受理官庁の名称及びあて名

日本国特許庁（RO/JP）

郵便番号 100-8915 電話番号 03-3592-1308

日本国東京都千代田区霞が関三丁目 4 番 3 号

様式 PCT/RO/106（2004 年 1 月）

権限のある職員

特許庁長官

図面は、特許協力条約に基づく国際出願等に関する法律施行規則第30条第1項第3号に規定する要件に適合しない。  
国際出願の図面について次の不備を発見した。

## I. 図面の用紙に関して

- a. ☐ 用紙が直接複製することができない。
- b. ☐ 用紙に折り目、しわ、裂け目がある。
- c. ☐ 用紙の両面が用いられている。
- d. ☐ 用紙が可撓性のある／丈夫な／白色の／滑らかな／光沢のない／耐久性のあるものではない。
- e. ☐ 図面が別の用紙で作成されていない。
- f. ☐ 用紙が所定のとり方ではない。
- g. ☐ 用紙の大きさが日本工業規格A列4番の大きさではない。(横21cm、縦29.7cm)
- h. ☐ 用紙の余白が所定のとり方ではない。(最少：上端2.5cm、左端2.5cm、右端1.5cm、下端1cm)
- i. ☐ 用紙に記載されている出願人又は代理人の書類記号が用紙の上端の余白の左隅であって上端から1.5cm以内に記載されていない。
- j. ☐ 出願人又は代理人の書類記号が12字を超えている。
- k. ☐ 用紙の使用することができる面又は使用した面の周囲に枠が記載されている。
- l. ☐ 用紙にアラビア数字により連続した番号が付されていない。(例：1/3、2/3、3/3)
- m. ☐ 用紙の番号が用紙の上端又は下端の中央に付されていない。
- n. ☐ 用紙の番号が余白内に記載されている。(余白には記載できない。h参照)
- o. ☐ 用紙に訂正／重ね書き／行間挿入／削除箇所が多く行われている。
- p. ☐ 用紙に複写の際のよごれがある。

## II. 図面に関して

- a. ☐ 図面が直接複製することができない。
- b. ☐ 不必要な記載事項がある。
- c. ☐ 図面の語句が翻訳された場合に、図面の線にかかるような記載がある。
- d. ☐ 耐久性のある、黒色の、十分に濃厚な濃墨等を用い、太さの均一な、かつ、明瞭な線で着色することなく作成されていない。
- e. ☐ 平行斜線によらない切断面がある。
- f. ☐ 縮尺による写真複製をしたときに容易に識別できない記載がある。
- g. ☐ 図式によらない尺度が記載されている。
- h. ☐ 簡潔かつ明瞭でない数字、文字、引出線がある。
- i. ☐ 製図用具を用いることなしに引かれた線がある。
- j. ☐ 図中の他の要素に対し妥当でない比率で記載した図がある。
- k. ☐ 0.32cm以下の大きさの数字又は文字がある。
- l. ☐ ローマ字及び慣習となっているギリシャ文字以外の文字の記載がある。
- m. ☐ 2以上の用紙に描かれた図であって単一の完全な図を得るように用紙を合わせたときに隠れる部分がある。
- n. ☐ 適切に配置されていない図がある。
- o. ☒ 個々の図に連続したアラビア数字による番号が付されていない。(注) 全図
- p. ☐ 用紙の番号と関係のある番号が付されている図がある。
- q. ☐ 明細書に用いていない引用符号が記載されている。
- r. ☐ 明細書に用いられている引用符号の記載がない。
- s. ☐ 異なった引用符号により表示された同一の部分がある。
- t. ☐ 個々の図が上下を正しく、相互に十分に離されていない。
- u. ☐ 図の上端が用紙の左側になるように描かれていない。

(注意)

手 続 補 正 書  
(法第6条の規定による命令に基づく補正)



特許庁長官 殿

1. 国際出願の表示 PCT/J P 2 0 0 4 / 0 0 0 5 2 1

2. 出願人 (代表者)

名 称 ヒューレット・パカード デベロップメント カンパニー  
エル. ピー.  
HEWLETT-PACKARD DEVELOPMENT  
COMPANY, L. P.  
あて名 アメリカ合衆国テキサス州 77070 ヒューストン 20555  
ステイト ハイウェイ 249  
20555 State Highway 249,  
Houston, Texas 77070 U.S.A.  
国 籍 アメリカ合衆国 US  
住 所 アメリカ合衆国 US

3. 代理人

名 称 特許業務法人 アイ・ピー・エス  
PATENT RELATED CORPORATION IPS  
代表者 早川 明 HAYAKAWA Akira  
あて名 〒221-0052  
日本国神奈川県横浜市神奈川区栄町5番地1  
横浜クリエーションスクエア12階  
12F, Yokohama Creation Square  
5-1, Sakae-cho, Kanagawa-ku,  
Yokohama-city, Kanagawa 2210052 JAPAN



4. 補正命令の日 20. 04. 04

5. 補正の対象 図面

6. 補正の内容 別紙の通り。  
全図に連続したアラビア数字による番号を付す。

7. 添付書類の目録

(1) 図面 16 / 16 頁

## 図 1

```
private void button5_Click()
{
    // DB への接続を行い、トランザクションを開始する
    DBConnection con = new DBConnection(接続情報);
    con.Open();
    DBTransaction tx = con.BeginTransaction();

    // DB への操作を行う
    AuthorsDataSet ds = new AuthorsDataSet();
    AuthorsDataSet.authorsRow row = ds.authors.NewauthorsRow();
    ...
    ds.authors.AddauthorsRow(row);
    AuthorsAdapter adp = new AuthorsAdapter();
    adp.Update(con, tx, ds.authors);

    // 次の DB 操作を行う
    ...

    // 通常はこのように複数の DB 操作を行う
    ...

    // トランザクションをコミットし、DB 接続を終了する
    tx.Commit();
    con.Close();
}
```

## 図 2

## UI 層のプログラム

```
private void button5_Click()
{
    App.PubsApp app = new App.PubsApp();
    app.InsertAuthorAndStore(textBox1.Text, textBox2.Text);
}
```

## AP 層のプログラム

```
public bool InsertAuthorAndStore(string au_id, string store_id)
{
    // DB への接続を行い、トランザクションを開始する
    DBConnection con = new DBConnection(接続情報);
    con.Open();
    DBTransaction tx = con.BeginTransaction();

    // DB 操作 # 1
    DB.Data.AuthorsData data1 = new DB.Data.AuthorsData();
    bool ret1 = data1.Insert(con, tx, au_id);

    // DB 操作 # 2
    DB.Data.StoresData data2 = new DB.Data.StoresData();
    bool ret2 = data2.Insert(con, tx, store_id);

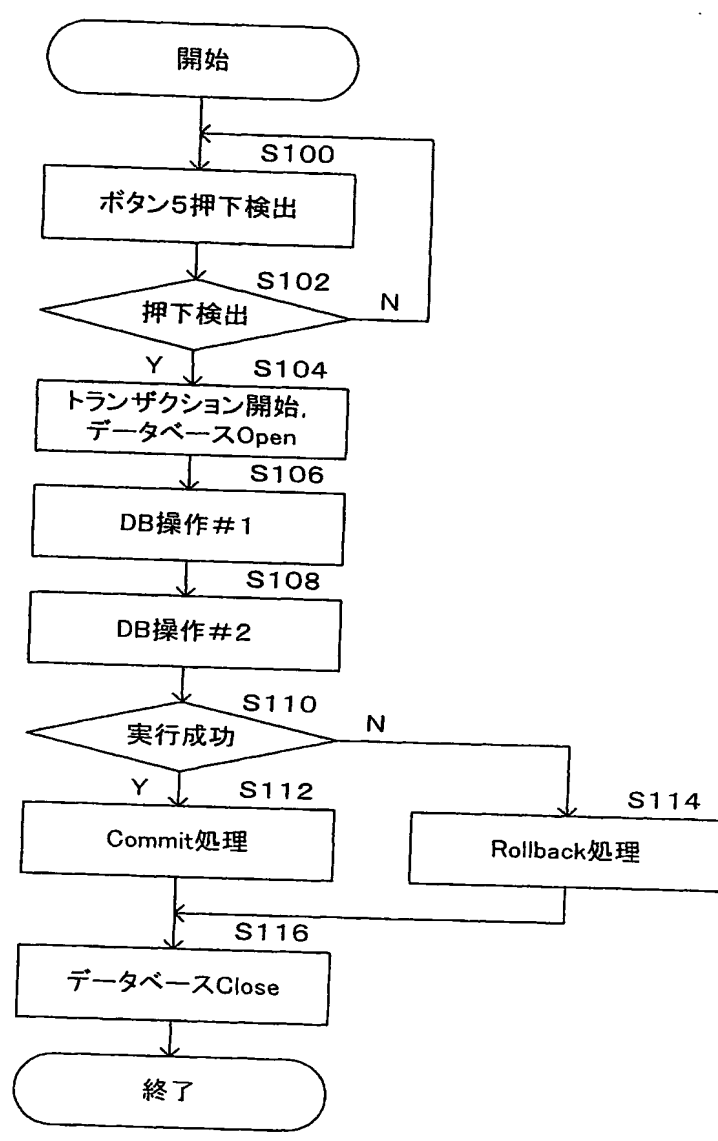
    // 全ての処理が成功したらトランザクションをコミットし DB 接続を終了する
    tx.Commit();
    con.Close();

    return true;
}
```

## DB 層のプログラム

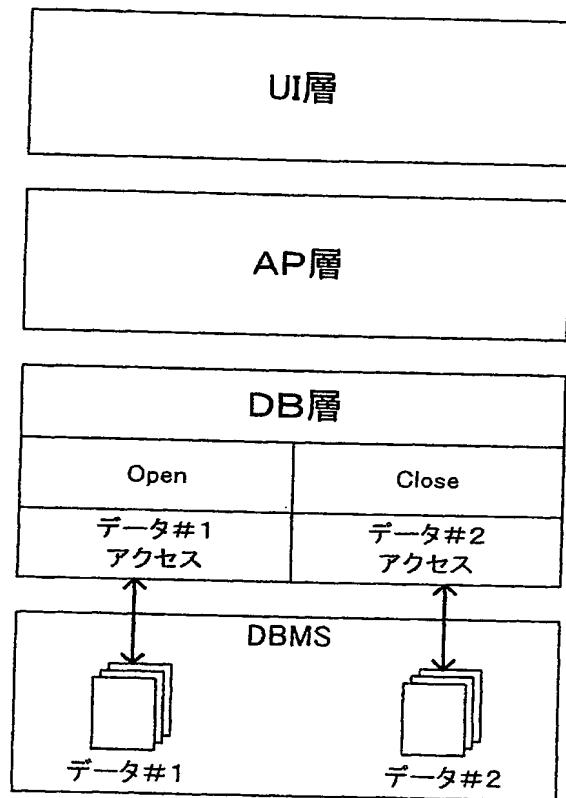
```
public bool Insert(DBConnection con, DBTransaction tx, string au_id)
{
    DB.AuthorsDB db = new DB.AuthorsDB();
    bool ret = db.Insert(con, tx, au_id);
    ...
    return ret;
}
```

図 3



S10

図 4



## 図 5

## UI 層のプログラム

```
private void button5_Click()
{
    App.PubsApp app = new App.PubsApp();
    app.InsertAuthorAndStore(textBox1.Text, textBox2.Text);
}
```

## AP 層のプログラム

[トランザクションの動作を指定する属性を指定]

```
public bool InsertAuthorAndStore(string au_id, string store_id)
{
    // DB 操作 # 1
    DB.Data.AuthorsData data1 = new DB.Data.AuthorsData();
    bool ret1 = data1.Insert(con, tx, au_id);

    // DB 操作 # 2
    DB.Data.StoresData data2 = new DB.Data.StoresData();
    bool ret2 = data2.Insert(con, tx, store_id);

    return true;
}
```

## DB 層のプログラム

```
public bool Insert(string au_id)
{
    SqlConnection con = new SqlConnection(strCon);
    con.Open();
    DB.AuthorsDB db = new DB.AuthorsDB();
    bool ret = db.Insert(con, tx, au_id);
    if (ret)
        ContextUtil.SetCommit();
    else
        ContextUtil.SetAbort();

    con.Close();
    return ret;
}
```



図 6

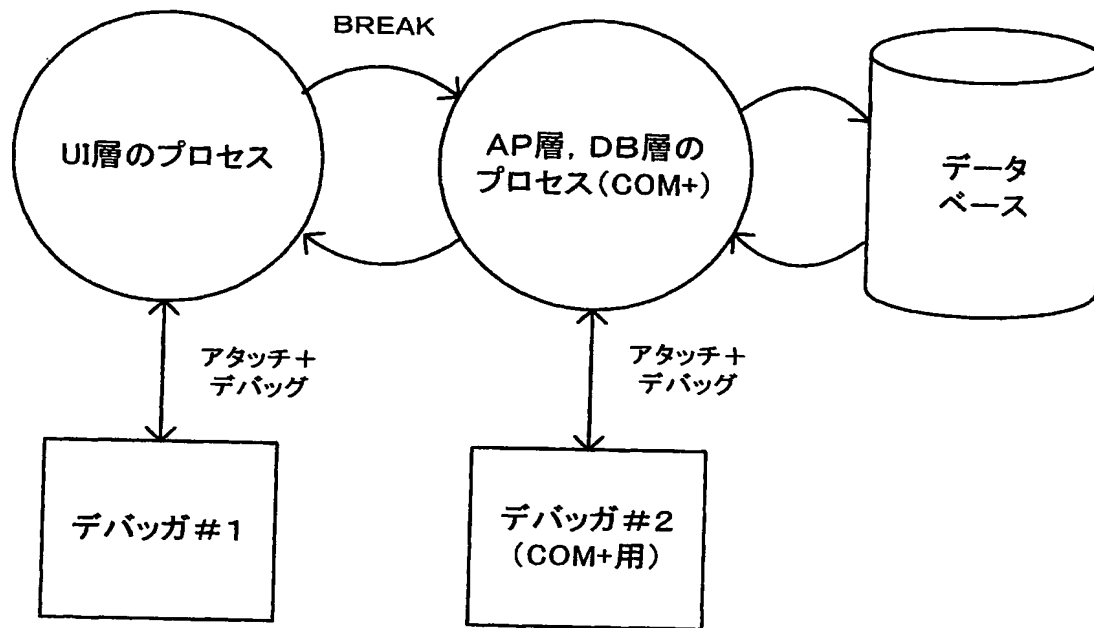


図 7

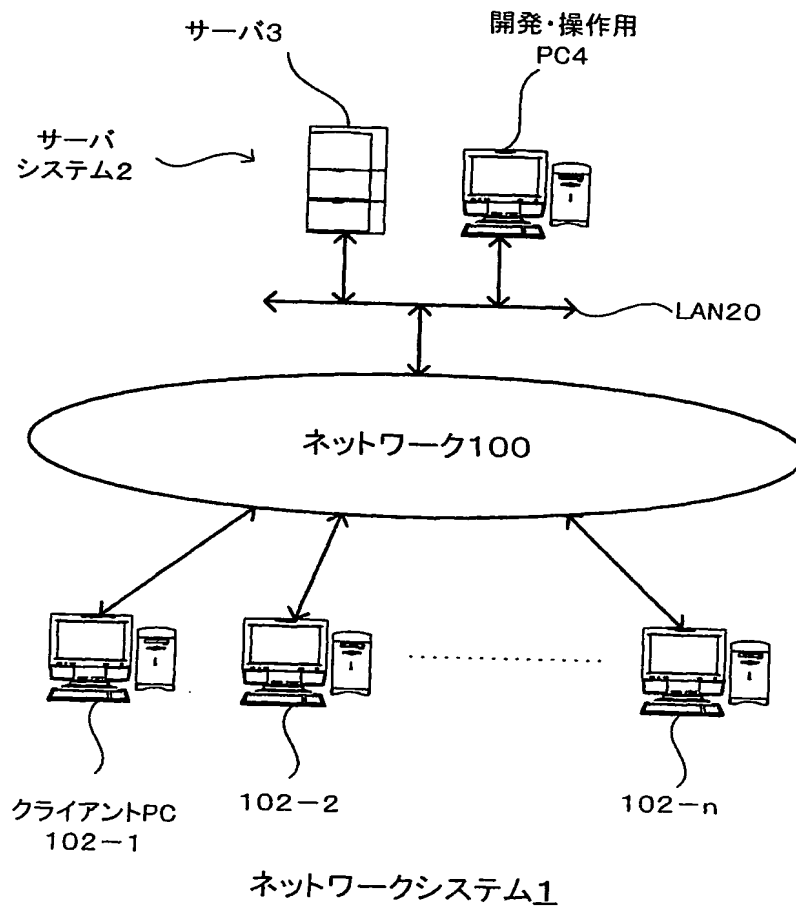
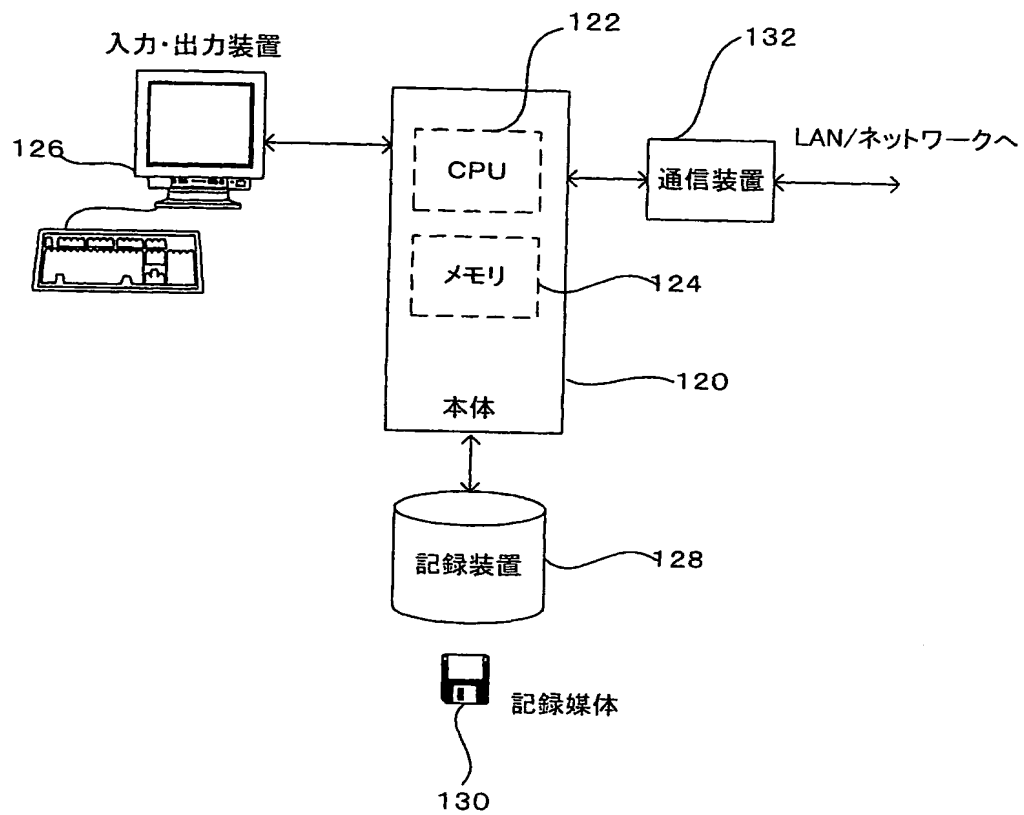
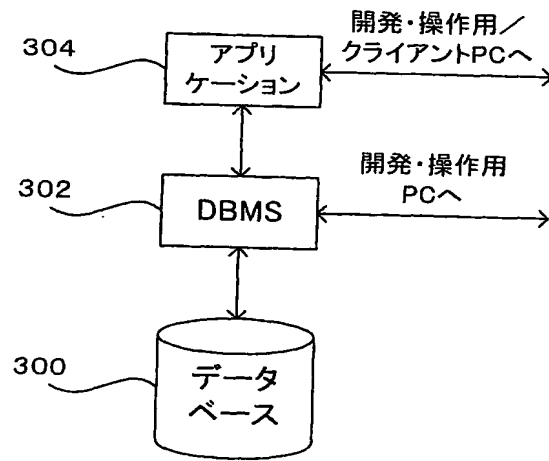


図 8



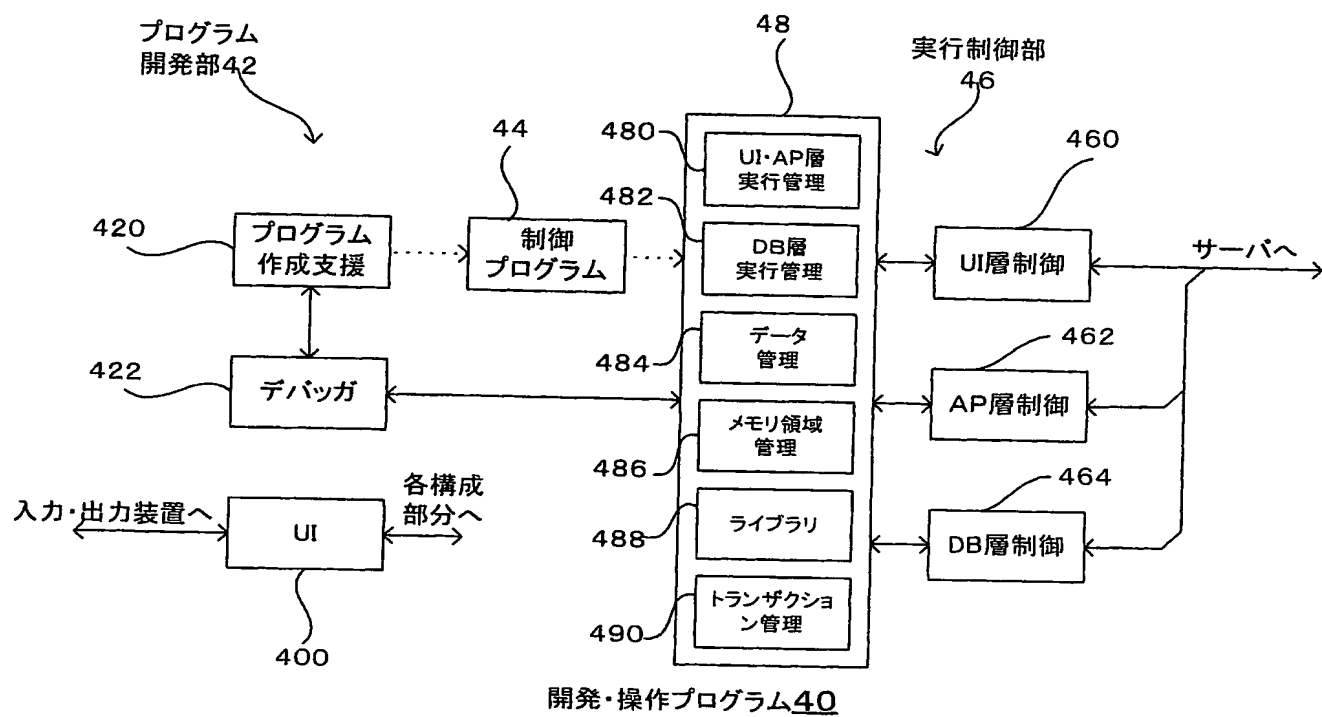
サーバ3,  
PC4, 102

図 9



サーバプログラム30

図 10



## 図 1 1

UI 層のプログラム

```
private void button5_Click()
{
    App.PubsApp app = new App.PubsApp();
    app.InsertAuthorAndStore(textBox1.Text, textBox2.Text);
}
```

AP 層のプログラム

[トランザクションの動作を指定する属性を指定]

```
public bool InsertAuthorAndStore(string au_id, string store_id)
{
    // トランザクションの定義
    using (new TransactionUnit())
    {
        // DB 操作 # 1
        DB.Data.AuthorsData data1 = new DB.Data.AuthorsData();
        bool ret1 = data1.Insert(con, tx, au_id);

        // DB 操作 # 2
        DB.Data.StoresData data2 = new DB.Data.StoresData();
        bool ret2 = data2.Insert(con, tx, store_id);
    }
    return true;
}
```

DB 層のプログラム

```
public bool Insert(string au_id)
{
    MySqlConnection con = new MySqlConnection(strCon);
    con.Open();
    MyTransaction tx = con.BeginTransaction();
    DB.AuthorsDB db = new DB.AuthorsDB();
    bool ret = db.Insert(con, tx, au_id);
    if (ret)
        tx.Commit();
    else
        tx.Rollback();

    con.Close();
    return ret;
}
```

図 1 2

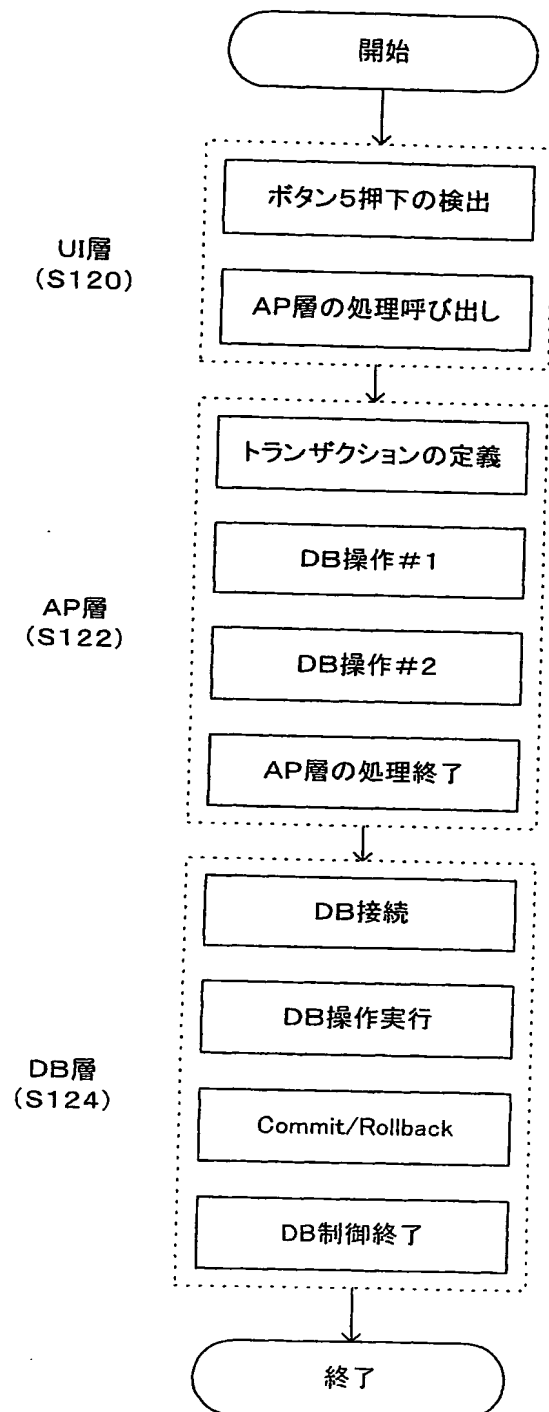


図 1 3

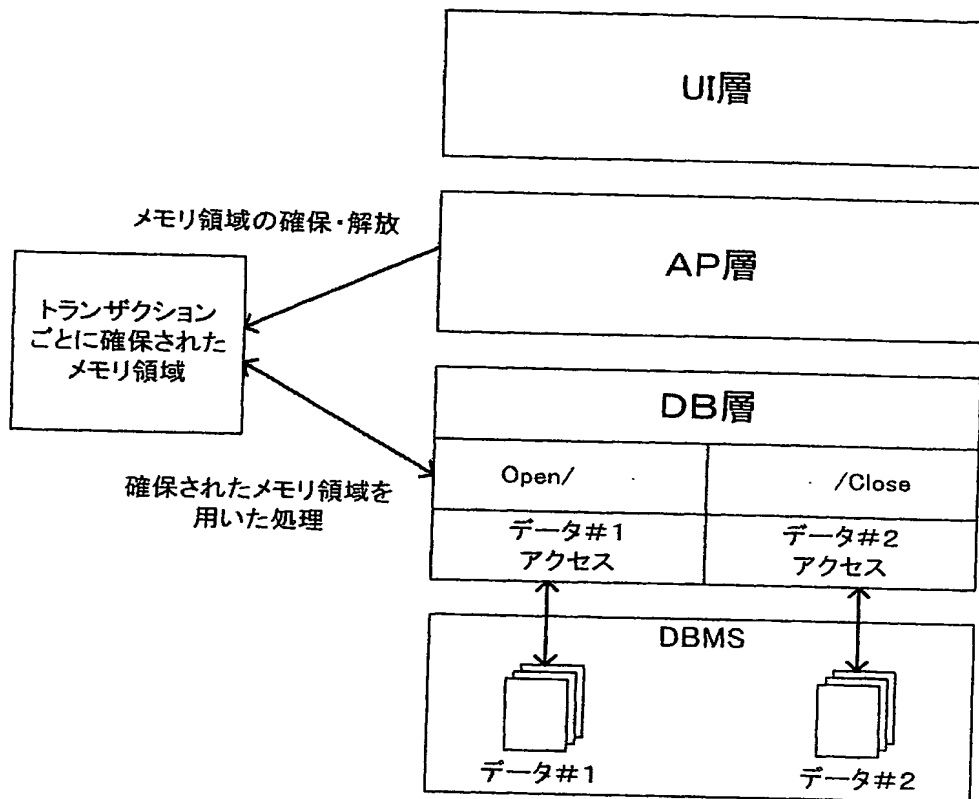




図 1 4

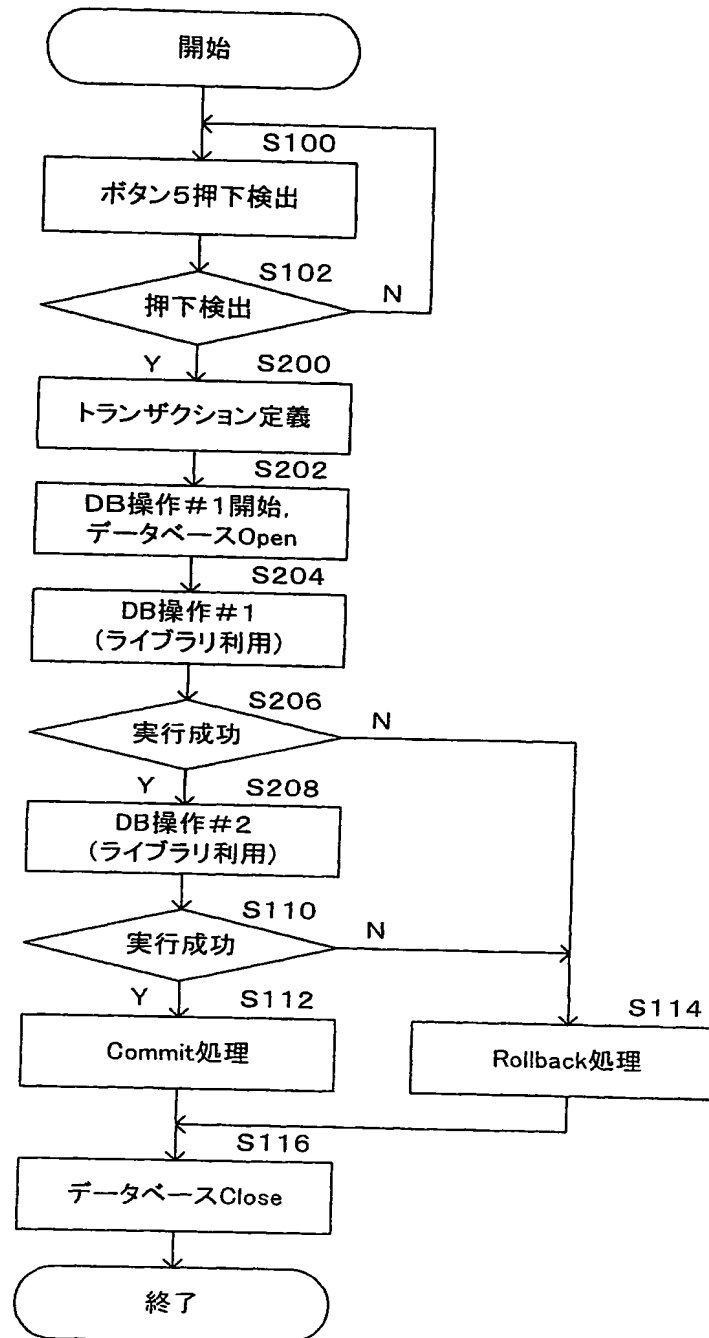
S20

図 1 5

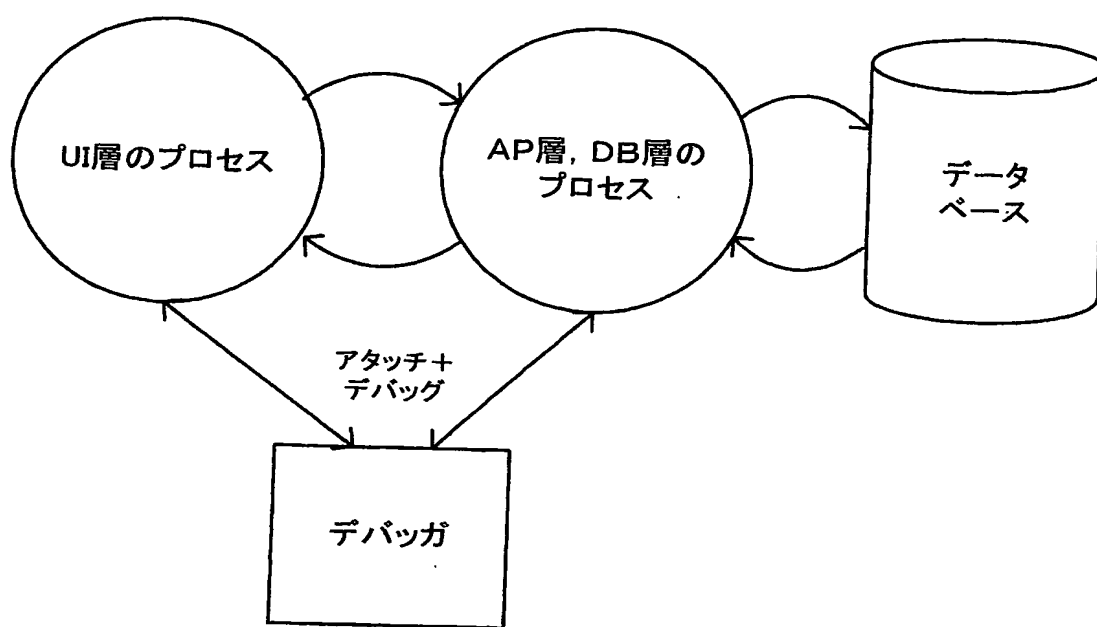


図 1 6

